

Custom design of CMOS low-power high-performance digital signal-processing macro for hard-disk-drive applications

by H. J. Shin
D. J. Pearson
S. K. Reynolds
A. C. Megdanis
S. Gowda
K. R. Wrenner

The design challenges and custom design techniques associated with low-power, small-area, high-performance CMOS digital signal-processing circuits for hard-disk-drive applications are presented. The advantages of custom design are demonstrated by an example custom digital FIR filter macro that provides substantial improvement in performance, area, and power dissipation over standard-cell implementations.

Introduction

A primary requirement of integrated circuits (ICs) in many signal-processing-related applications is maximum performance at minimum power consumption and cost.

This requirement presents significant challenges in making the right set of decisions in all phases of IC design, including IC technology, algorithm, architecture, logic family, circuit technique, layout, and design methodology. Generally, for a fixed design choice, the speed or performance of a digital IC can be improved by increasing the device size and, accordingly, the power and area consumption, as illustrated in **Figure 1**. Eventually, the speed saturates even though more power is expended, because the circuit approaches an intrinsic speed limit when the wiring capacitance becomes a minor fraction of the device capacitance. Therefore, an optimum high-performance design point is one at which the design curve begins to saturate. Starting from this point, to obtain higher performance at lower power, the curve must be shifted upward and to the left, as indicated by the arrow in

©Copyright 1995 by International Business Machines Corporation. Copying in printed form for private use is permitted without payment of royalty provided that (1) each reproduction is done without alteration and (2) the *Journal* reference and IBM copyright notice are included on the first page. The title and abstract, but no other portions, of this paper may be copied or distributed royalty free without further permission by computer-based and other information-service systems. Permission to *republish* any other portion of this paper must be obtained from the Editor.

0018-8646/95/\$3.00 © 1995 IBM

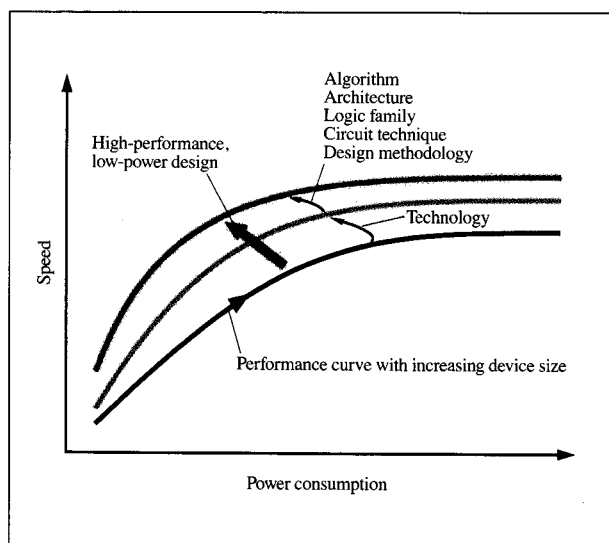


Figure 1

Design curves and techniques for high-performance, low-power digital ICs.

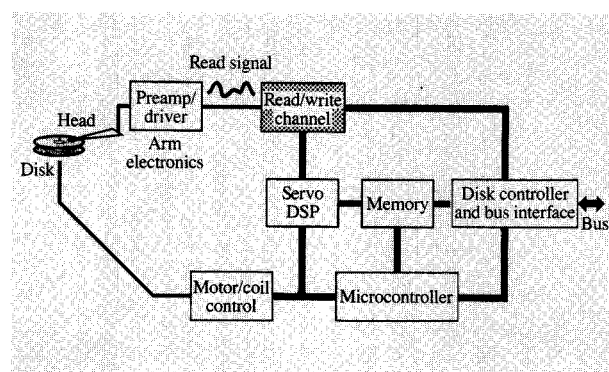


Figure 2

Hard-disk-drive system.

Figure 1. This means implementing the design in a more advanced technology (typically, scaled to a smaller feature size) and/or redesigning with a better algorithm, architecture, logic, circuit, and/or layout. Although enhancing performance using a better technology is relatively simple and easy from the design point of view, the implementation of a new technology is usually very expensive. It is far more economical, though challenging, to exploit the capability of a given technology, improving the existing design where possible. Generally, the improvement involves custom design techniques that often

are limited to layout but can be extended to other aspects of design as well. Custom design allows optimization at every level of the implementation and also removes other constraints. For example, special custom circuits and layout may permit certain macro functions to be realized in much smaller areas and to operate much faster, which in turn may permit the application of new architectures or algorithms. One disadvantage of custom design, however, is longer design turnaround time, which can slow the product cycle. In this paper, the capability and leverage of custom design are demonstrated through the design of a CMOS digital signal-processing FIR filter macro.

Hard-disk-drive channel

As the primary storage devices in computer systems ranging from mainframes to notebook personal computers, modern hard-disk drives are faced with ever-increasing demands of high storage density, low cost, low power consumption, small size or form factor, high data rate, high reliability with shock resistance, and fast design or product cycle. Such demands are more aggressive than those made of most other products. All specifications must be improved simultaneously by a factor of about 2 in each generation of hard-disk drives, with a new generation appearing every one and a half to two years. Consequently, all components in a typical disk-drive system, such as disk or storage medium, head, and electronics, as shown in **Figure 2**, require continuous re-engineering. Especially challenging is the design of electronics in the data write/read path or channel—usually a performance bottleneck—to satisfy the bit density and data rate requirements while minimizing the cost/performance and power/performance ratios. The characteristics of the channel often resemble those of typical serial-data digital communication channels. As the serial bit stream is written and read, it becomes distorted by the magnetic disk and head and is degraded by intersymbol interference and noise. Therefore, like communication circuits transceiving at maximum link bandwidth, the channel electronics usually provide pre-amplification of the distorted signal, automatic gain control, anti-alias filtering, clock recovery and generation, sophisticated signal processing, encoding and decoding, and multiplexing and demultiplexing of the data.

Design of high-performance, low-power, and low-cost channel electronics requires careful selection of the signaling and detection algorithm, signal processing technique, technology, functional block architecture, circuit technique, and design methodology. For example, state-of-the-art read/write channel ICs employ the partial-response signaling, maximum-likelihood detection (PRML) algorithm instead of the peak detection algorithm, analog/digital mixed signal processing with a combination of bipolar and CMOS technologies or a bi-CMOS

technology, a flash-type analog-to-digital converter (ADC) for digitizing the degraded bit-stream signal into multi-bit codes, a digital finite-impulse-response (FIR) filter for processing the codes, and bipolar custom analog circuitry together with CMOS standard-cell digital logic [1-5]. A simplified block diagram of such a PRML channel IC with a 6-bit ADC and a digital FIR filter is shown in **Figure 3**. The remaining analog circuitry typically consists of a variable-gain amplifier (VGA), an anti-aliasing analog filter, a phase-locked loop (PLL) for clock recovery, digital-to-analog converters (DACs), a servo detector, and a write pre-compensation circuit. The digital portion includes a Viterbi detector, a decoder, an encoder, and timing and gain control logic. PRML with analog-only sampled-data signal processing is also a good alternative for reducing the chip power and size [6], though not as accurate or flexible. Compared to digital signal processing, analog signal processing requires a very small number of circuit elements such as sampled analog delay lines and analog multiplier/adders with moderate accuracy and matching in terms of linear resolution.

Once the algorithm, signal-processing method, and architecture are optimized, channel IC performance can be improved most easily by scaling the design of an existing technology to smaller feature sizes and faster devices. However, because of the analog/digital nature of the design, the technology to be scaled is actually a set of mixed-signal technologies whose development is both difficult and costly. The technology itself will also be quite expensive. An effective way to increase the channel IC speed while staying with a low-cost technology is to use custom design, even for digital macros. Beyond the obvious advantages of optimized custom design (i.e., less chip area and power and/or higher performance), noise generated and coupled from custom digital macros is less than that from standard-cell macros; this alleviates somewhat the noise isolation problem associated with mixed-signal design. However, because the design time, cost, and flexibility are compromised, custom design would best be limited to high-volume or high-profit ICs and to high-leverage critical macros whose functions are relatively stable or standardized. In the PRML channel IC shown in **Figure 3**, the digital FIR filter is an example of such a macro, suitable for customization. The following sections describe the custom design of this macro.

Digital FIR filter architecture

The digital FIR filter equalizes the channel frequency characteristics. It is essentially an adaptive—or programmable in typical applications where the adaptation rate is very low—transversal filter with a tapped delay line and associated tap weights that are updated as necessary based on measured characteristics of the channel [7]. Compared to an analog FIR filter, the digital FIR filter

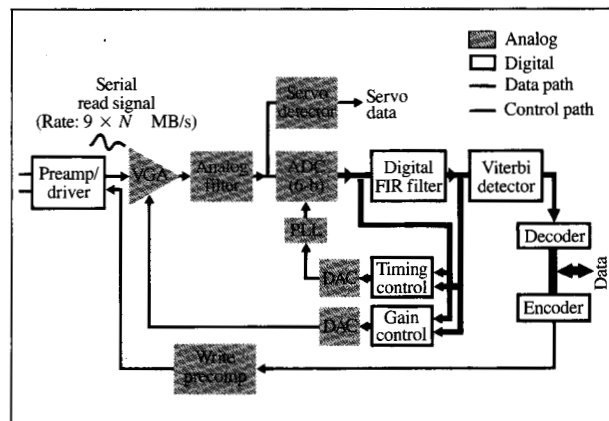


Figure 3

Partial-response maximum-likelihood (PRML) channel IC with analog/digital mixed-signal processing.

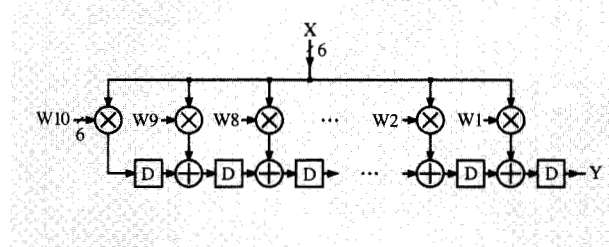


Figure 4

10-tap, 6-bit digital FIR filter in transpose architecture.

offers flexibility in changing the filter characteristics as well as accuracy (meaning that it is insensitive to noise and nonlinearity). Its input is multi-bit data from the analog-to-digital converter, representing quantized samples of the serial bit-stream signal through the channel. Filter accuracy is determined by the number of taps and the number of bits representing the input data and tap weights. Larger numbers improve the accuracy, but they increase the chip area and power consumption. The set of optimum tap number and data word width adopted for this design is 10 and 6-bit, respectively [2].

The 10-tap, 6-bit FIR filter represented as a generic signal-flow diagram is shown in **Figure 4**. A straightforward implementation of the depicted structure (the so-called transpose architecture) would require ten multipliers, ten adders, and ten registers. The multiplier and adder should handle most of the 12-bit product term to ensure the accuracy of the final output. This particular implementation simply repeats the basic multiplier-adder-register

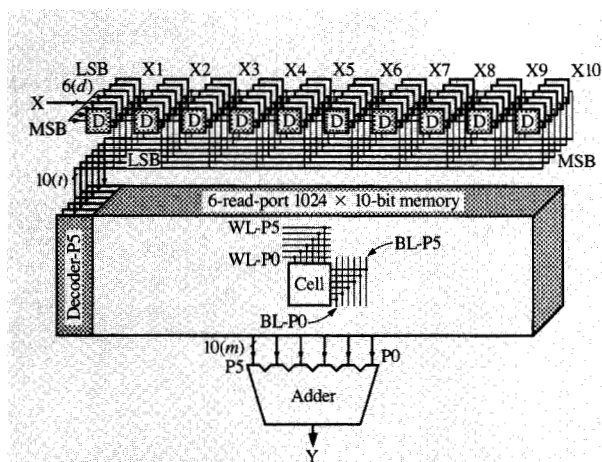


Figure 5

10-tap, 6-bit digital FIR filter with distributed-arithmetic (DA) architecture.

building block, so redesigning for a different tap number would be an easy task. Another point to note is that the area and power of the macro are linearly proportional to the tap number. Thus, the area and power of each building block must be minimized. However, typical high-performance multipliers and adders—employing Booth encoding, carry-save adders, Wallace-tree partial product reduction, and carry-lookahead addition techniques—would require a large number of devices and wires.

Because the tap weights may be treated as constants during normal operation, the filter can be implemented with the distributed-arithmetic (DA) architecture [8] that replaces the ten multiplication/additions of the ten tap operands with one combined, multi-port (6-read-port in this case) memory-read/addition, as shown in simplified form in **Figure 5**. The 6-read-port memory (or register file) stores precalculated partial sums of all ten tap weights. To write this memory, one additional write port is needed. Each read port corresponds to a bit position of the data [i.e., port 0 to the least significant bit (LSB), ..., and port 5 to the most significant bit (MSB)], and the memory word is accessed by a 10-bit address assembled with every corresponding bit of the 10-tap data. For example, if all LSBs of the 10-tap data are equal to 1, the sum of all ten weights would be stored in word location 1023 of port 0. If the MSBs of the 10-tap data are 0000000001, only the tenth weight would be stored in word 1 of port 5. In general, if implemented as is, the memory word size would be 2^t (where t is the number of taps), its port number would be equal to the data bit width d , and the memory data width

m representing the accuracy of the partial sum would be $d + \log_2 t$. The adder sums all the $d \times m$ -bit outputs from the memory simultaneously. Therefore, the DA architecture would require a large memory and a large, fast adder. If t is reduced, the delay, area, and power of the memory will decrease rapidly, while those of the adder will decrease marginally. Note that if t is large, the memory size would be too big to make this architecture useful compared to the transpose architecture with t multiplier/adders, unless the multi-port memory cell is sufficiently small.

One way to reduce the memory size is to break the memory into two; e.g., data in odd tap positions would access one memory and those in even tap positions would access the other. The word size of each memory and total memory would then be $2^{t/2}$ and $2^{1+t/2}$, respectively, and $m = d + \log_2(t/2)$. In the 10-tap case, the total word size is 64—a drastic (16 $\times$) reduction from 1024. Note that this technique requires an adder up to twice as large for summing $2 \times d \times m$ -bit memory outputs. However, the overall reduction in size is significant. There are other techniques for exploiting the symmetric nature of tap weights and associated signed-digit offset-binary numbers in order to reduce hardware size even further [8]. One such technique involves exclusive-OR (XOR) operation on all memory address bits except the MSB, and on the memory output bits, with the MSB of the address. This allows removal of the MSB and reduces the memory size by a factor of 2 (to 32) and corresponding m (to 8). In all, the combined even/odd addressing and XOR techniques require two 6-read, 1-write-port 16×8 memories, each of which is accessed by a 4-bit address that represents four XORed LSBs of even- or odd-tap outputs of the 10-tap, 6-bit input shift register (or delay line). They also need an adder to sum the 96-bit ($= 2 \times 6\text{-port} \times 8\text{-bits/port}$) XORed output of the memories and produce the final one-byte result. Using these memory-size reduction methods as well as custom-designed memory circuits for the design of a 10-tap, 6-bit FIR filter, the DA architecture is slightly advantageous in terms of area and power consumption at a given speed, compared to the transpose architecture. For a smaller number of taps, because the memory size is reduced rapidly with the decreasing tap number, the DA architecture would be even more attractive.

The area-efficient architecture of the digital FIR filter incorporating all the techniques above (which is similar to the one in [2]) should be tuned to satisfy the performance requirement of the filter as well. In this DA architecture, the data computation path consists of the memory access and the addition. Between the two, the 96-bit input addition would take much longer than the memory access because the memory word size is only 16. Thus, it is desirable to speed up the addition using a combination of fast adders such as the Wallace-tree carry-save adder and

the carry-lookahead adder. Once the data-path delay is minimized, because the data flow is quite homogeneous and unidirectional along the path without any feedback loop, the cycle time of the filter can be set to meet the data-rate requirement by introducing pipelining, i.e., breaking the path and inserting pipeline registers. Here, the number of required pipeline stages is determined as the total delay divided by the clock cycle necessary for the data rate specified.

Custom digital FIR filter design

The required worst-case data rate of this 10-tap, 6-bit FIR filter is derived from the channel performance specification, which is 20 MB/s in this example. For this performance, the signal read and amplified on the channel is a serial bit stream with a rate of 180 Mb/s (for 8/9 coding [2]). Because the FIR filter operates on samples of this signal digitized by the A/D converter, its operating frequency should be at least 180 MHz. To handle this rate while minimizing the power, area, and cost, proper selection of the technology as well as adoption of custom design methodology is necessary in addition to the choice of the DA architecture discussed above. The technology selection is constrained by the operating environment and mixed-signal nature of the PRML channel IC. Even though more advanced (or aggressively scaled) technologies would have performance, power, and area advantages, they would not allow the integration of mixed-signal circuits because of the lower supply voltage required by these technologies or the lack of good analog devices available in them. The technology chosen in this example is a bi-CMOS technology with 0.5- μm effective channel length and 3.3-V power supply [9]. This technology, though not state-of-the-art in feature size and gate delay, is mature enough to be inexpensive and offers high-transconductance bipolar transistors and resistors for the analog circuits on the chip. With this, however, the challenge is to deliver the required performance of critical macros such as the FIR filter at the worst-specified operating environment, i.e., a supply voltage as low as 2.82 V (-10% tolerance and 150-mV additional line drop) and a nominal/worst-case temperature of 50/100°C, respectively. In fact, a CMOS design based on standard-cell methodology shows that the filter functions only up to 126 MHz. To achieve the specified performance, custom techniques at all design levels (logic, circuit, and layout) are found to be effective and essential. The custom design further makes it possible to implement the digital filter circuit for the desired speed using only CMOS devices, although bi-CMOS circuits or bipolar devices available in the technology can be used to boost the speed. An all-CMOS circuit is advantageous because it can be extended and remapped easily to future technologies.

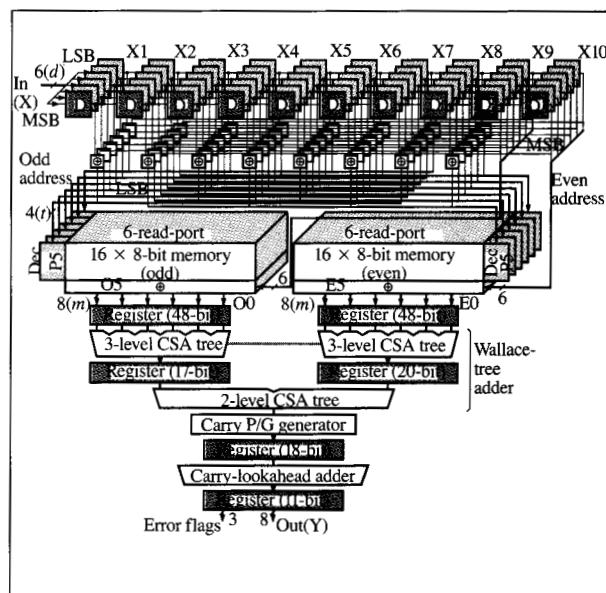


Figure 6

DA-architecture 10-tap, 6-bit digital FIR filter with minimized hardware.

As the first step in custom design, implementation details of the DA filter architecture reflecting practical logic and circuit design considerations have been finalized, as shown in Figure 6. On the basis of preliminary circuit sizing and delay assessment, the filter has been broken into four pipeline stages whose delays are more or less balanced against one another, meeting the 180-MHz clock cycle requirement. Note that the pipeline registers are placed to balance the delays, without being constrained to the functional boundaries. The first pipeline stage corresponds to the memory access, including the XORing. The 96-bit output from the memories, latched in the first 96-bit pipeline register, is connected to the Wallace tree; the 48 bits from the memory with even addresses go to one branch and the 48 bits from the memory with odd addresses go to the other branch. The Wallace tree consists of five levels of carry-save adders that are broken into a three-level tree in the second pipeline stage and a two-level tree in the third stage. Each carry-save adder is a differential 3-to-2 counter, or full adder, and the wiring inside the tree is illustrated in Figure 7. Of the 48 bits for each branch, eight bits from the LSB port (port addressed by the LSBs of the shifted input data at the even or odd tap positions) are wired to the LSB locations of the tree, and eight bits from the MSB port, to the MSB locations. The second and third pipeline registers are 37 bits and 18 bits wide, respectively. The third pipeline also contains the front part of the 8-bit carry-lookahead adder, i.e., the carry

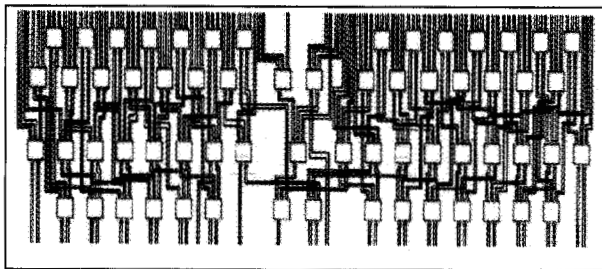


Figure 7

Wallace-tree carry-save adder wiring diagram: Each box represents a full adder with differential inputs/outputs.

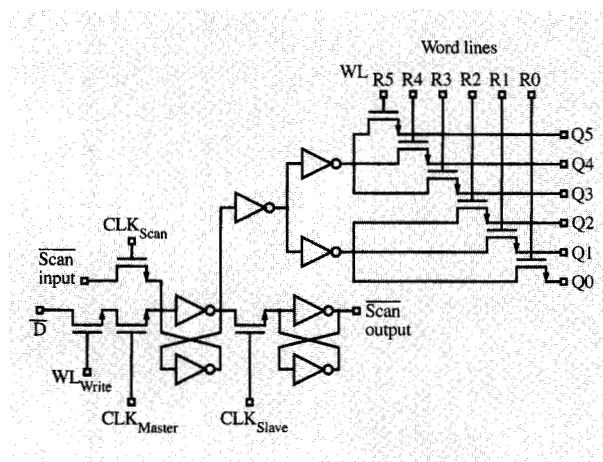


Figure 8

Memory cell circuit with six read ports and one write port.

propagate/generate logic. The last stage covers the main part of the carry-lookahead adder, which produces the final sum and any error flags. Rounding of the LSB for the final sum is done in the second pipeline stage.

The functional behaviors of the sub-macros such as the input shift register, memory, Wallace-tree adder, and final adder have been modeled, simulated, and verified individually as well as collectively with pseudorandom test vectors. In addition, the overall filter function and logic have been verified against a reference input data stream and corresponding set of tap weights by looking at the filtered output. The test coverage of the filter is enhanced by two LSSD (level-sensitive scan design) scan-chains: one through the memory and the other through the remaining logic. To control the timing skew and overlap of different

clock phases precisely at this high speed, the latch clocks are generated locally from a master clock, and each set of local clocks is limited to drive only six to nine latches.

The circuits have been custom-designed by applying proper topologies and techniques, and their performance, power dissipation, and area have been tuned through optimization of every transistor size using a circuit simulator. For the input shift registers and pipeline registers, an LSSD-compatible, differential, edge-triggered master/slave latch circuit is used. This latch, having true and complementary inputs/outputs, interfaces well with the memory and Wallace tree, with minimal set-up time and delay. Its structure, identical for both the master and the slave sections, resembles the standard CMOS static memory cell. The multi-port (6-read and 1-write) memory cell is a single-ended, buffered latch that minimizes the read-access delay associated with large bit-line capacitance, as depicted in **Figure 8**. The master latch in the memory stores the data written through the two input pass gates in series. The data are read from the master section through the two-stage buffer and *n*-channel output pass gates selected by the read word-lines. In this cell, the slave latch is used only during LSSD scanning. For the carry-save adder, a conventional full-complementary transmission-gate logic circuit with complementary signals is chosen because it performs better than alternative circuits under the worst-case set of conditions and electrical parameters, even though wiring complexity is doubled compared to single-ended circuits. For the final carry-lookahead adder, standard static and transmission-gate CMOS logic circuits are selected for their performance advantage.

The layout of the filter macro has been designed with custom methodology. First, all of the cells, including latch, memory, and full-adder cells, are hand-honed for better integration density. **Figure 9** shows the layout of the memory cell, whose size is $62 \mu\text{m} \times 43 \mu\text{m}$. The filter, containing about 25000 transistors, is wired with only two metal layers so that it can be placed in a chip while preserving 100% wiring porosity on the third metal layer. It is loosely packed with a net circuit area of 4 mm^2 , excluding the I/O pads and receiver/driver circuits, as shown in **Figure 10**. This area is smaller by a factor of 2, compared with a standard-cell-designed filter with the same DA architecture in the same technology. Here, data flow upward, beginning at the input shift register located at the bottom of the figure. Note that the two 16×8 memories in the middle occupy a major portion of the area. The layout has been proven against the logic and circuit design to the extent that it can be verified by a layout-to-logic verification tool. Also, the delay of critical paths extracted from the layout has been checked against the design specifications through final circuit simulations.

Results and comparison

An experimental chip containing the custom-designed FIR filter has been fabricated and packaged using 0.5- μm bi-CMOS technology [9]. Functional test vectors have been generated by the logic design tool. From the first samples of the chip, the filter macro successfully passes all functional tests applied directly at the inputs (for the data and memory contents) or via the two LSSD scan-chains. The filtered output observed for the reference input data and tap weights matches the expected simulation results. As packaged, at room temperature, the chip functions up to 290, 270, and 250 MHz with a 3.6, 3.3, and 3.1-V power supply, respectively, without any errors in ten million test vectors. For 3.6 V, the power consumption is about 690 mW at 200 MHz and 1 W at 290 MHz. It drops to 580 mW at 3.3 V and 200 MHz. As expected for CMOS digital circuits, the power is proportional to clock frequency and has quadratic dependency on the supply voltage. The most critical signal path of the filter, identified by analyzing the first errors observed for a higher-speed clock, is the error-flag generation circuit in the carry-lookahead adder of the final pipeline stage. These measurement results are consistent with the simulation.

This custom-designed digital FIR filter more than satisfies the 180-MHz clock rate specified for the hard-disk-drive read-channel application. Compared to the filter with the same architecture, designed in the same technology but with a standard-cell methodology, it improves the attainable data rate by a factor of about 1.5 and the power consumption by a factor of 2. Combined, the custom filter has a nearly six-times-better power-delay-area product. Because the power reduction comes from decreased total switching and coupling capacitance, supply noise due to switching current as well as circuit noise due to capacitive coupling can be expected also to decrease by roughly the same factor. Thus, this custom filter would integrate more easily with analog circuits in a mixed-signal PRML channel IC than would the standard-cell-designed filter.

Conclusion

For IC applications in which power, area, and performance are all critical, e.g., the hard-disk-drive channel, custom digital design is one of the essential means for improving the specifications while reducing the cost, without relying on a more advanced technology. Custom design approaches at all levels of the hierarchy—involving appropriate selection as well as optimization of the algorithm and architecture, logic families for different subfunctions, circuit techniques and device sizes, and layout style—contribute collectively toward achieving the goal. In the 10-tap, 6-bit digital FIR filter example, custom techniques have made the DA architecture more attractive than the transpose architecture. Finally, customization at

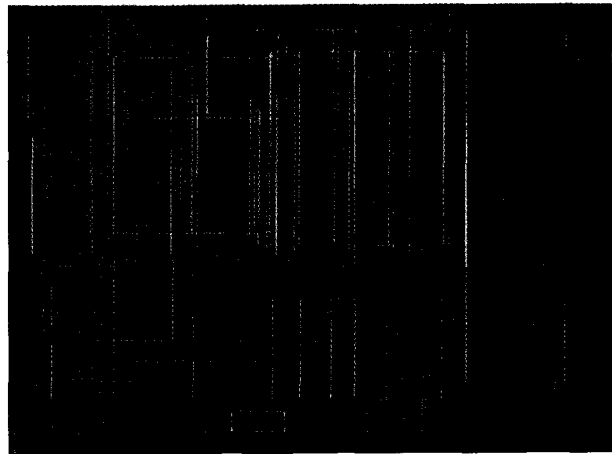


Figure 9

Layout of memory cell with six read ports and one write port: Size is $62 \times 43 \mu\text{m}^2$.



Figure 10

Layout of 10-tap digital FIR filter macro.

the logic, circuit, and layout levels has demonstrated a significant improvement in the power-delay-area product over a standard-cell design.

Acknowledgment

The authors would like to thank W. Pence, M. Arienzo, K. Toh, M. Chen, J. Belleson, S. Hamdani, R. Lynch,

G. Kerwin, and M. Kerr for their encouragement and support. Special thanks go to R. Galbraith and L. Thon for the technical discussions and to M. Immediato for testing help.

References

1. T. J. Schmerbeck, R. A. Richetta, and L. D. Smith, "A 27MHz Mixed Analog/Digital Magnetic Recording Channel DSP Using Partial Response Signaling with Maximum Likelihood Detection," *ISSCC Digest of Technical Papers*, pp. 136-137 (February 1991).
2. R. A. Philpott, R. A. Kertis, R. A. Richetta, T. J. Schmerbeck, and D. J. Schulte, "A 7 Mbyte/s (65 MHz), Mixed-Signal, Magnetic Recording Channel DSP Using Partial Response Signaling with Maximum Likelihood Detection," *IEEE J. Solid-State Circuits* **29**, No. 3, 177-184 (March 1994).
3. David R. Welland, Sandra M. Phillip, Ka Y. Leung, G. Tyson Tuttle, Scott T. Dupuie, Douglas R. Holberg, Randall V. Jack, Navdeep S. Sood, Kent D. Anderson, Alan J. Armstrong, Richard T. Behrens, William G. Bliss, Trent O. Dudley, William R. Foland, Jr., Neal Glover, and Larry D. King, "A Digital Read/Write Channel with EEPRA Detection," *ISSCC Digest of Technical Papers*, pp. 276-277 (February 1994).
4. Davy Choi, Richard Pierson, Fredrick Trafton, Benjamin Sheahan, Venugopal Gopinathan, Glenn Mayfield, Indumini Ramnuthu, Srinivasan Venkatraman, Vivek Pawar, Owen Lee, William Giolma, William Krenik, William Abbott, and Ken Johnson, "An Analog Front-End Signal Processor for a 64Mb/s PRML Hard-Disk Drive Channel," *ISSCC Digest of Technical Papers*, pp. 282-283 (February 1994).
5. W. L. Abbott, H. C. Nguyen, B. N. Kuo, K. M. Ovens, Y. Wong, and J. Casasanta, "A Digital Chip with Adaptive Equalizer for PRML Detection in Hard-Disk Drives," *ISSCC Digest of Technical Papers*, pp. 284-285 (February 1994).
6. R. G. Yamasaki, T. Pan, M. Palmer, and D. Browning, "A 72Mb/s PRML Disk-Drive Channel Chip with an Analog Sampled-Data Signal Processor," *ISSCC Digest of Technical Papers*, pp. 278-279 (February 1994).
7. A. V. Oppenheim and R. W. Schaffer, *Digital Signal Processing*, Prentice-Hall, Inc., Englewood Cliffs, NJ, 1975.
8. F. Dolivo and W. Schott, "Distributed Arithmetic Implementation of a Programmable Digital FIR Filter," *Technical Report SP011/1987*, IBM Zurich Research Laboratory, August 1987.
9. B. J. Gross, H. Chuang, and T. Schmerbeck, "State-of-the-Art Analog BiCMOS," invited paper, IEEE GaAs IC Symposium, San Jose, 1993.

Received October 11, 1994; accepted for publication November 28, 1994

Hyun J. Shin *IBM Research Division, Thomas J. Watson Research Center, P.O. Box 218, Yorktown Heights, New York 10598 (HJSHIN at YORKTOWN)*. Dr. Shin received the B.S. degree in electronics engineering from Seoul National University in 1978, the M.S. degree in electrical engineering from the Korea Advanced Institute of Science and Technology, Seoul, Korea, in 1980, and the Ph.D. degree in electrical engineering from the University of California, Berkeley, in 1988. From 1980 to 1983, he was with the Korea Institute of Electronics Technology as a researcher and senior researcher working on MOS single-chip microcomputers. Since 1988, he has been with the IBM Thomas J. Watson Research Center as a research staff member working on high-performance, low-power logic circuits and memories for computers and communications. Specifically, he has been concentrating on bi-CMOS logic circuits, high-bandwidth bipolar crosspoint switches, low-power high-speed bipolar logic circuits, ultrafast cache memories, custom bipolar and CMOS processors, on-chip voltage regulators for low-voltage CMOS, flash EPROM, custom CMOS digital signal-processing macros for hard-disk drive and video decoder applications, CMOS high-speed serial-link ASICs for ATM networks, and low-power design techniques. He is currently manager of the Digital Communication IC Design group. Dr. Shin is a senior member of IEEE.

Dale J. Pearson *IBM Research Division, Thomas J. Watson Research Center, P.O. Box 218, Yorktown Heights, New York 10598 (PEARSON at YKTMV)*. Mr. Pearson is a research staff member in the Digital Communication IC Design group. He received his B.S. degree in chemistry from Texas Lutheran College in 1979 and an M.S. degree from the University of Wisconsin, Madison, in 1981. After working in the IBM General Technology Division and the General Electric Aerospace Control Systems Department, in 1984 he joined the IBM Research Division at Yorktown Heights, where he has worked on the development of on-chip interconnects, process technology for low- T_c superconducting circuits, and custom integrated circuit design. Mr. Pearson is a member of the American Physical Society.

Scott K. Reynolds *IBM Research Division, Thomas J. Watson Research Center, P.O. Box 218, Yorktown Heights, New York 10598 (SCOTT at YORKTOWN)*. Dr. Reynolds received a B.S.E.E. degree from the University of Michigan in 1983 and an M.S.E.E. degree from Stanford University in 1984. From 1984 to 1987 he did thesis research on GaAs materials and devices; he received the Ph.D.E.E. degree from Stanford University in 1987. In 1988 Dr. Reynolds joined the IBM Thomas J. Watson Research Center, where he worked on advanced interconnect technology. In 1992 he joined the Digital Communication IC Design group, where he has designed high-speed signal-processing circuits. He is currently a research staff member.

Andrew C. Megdanis *IBM Research Division, Thomas J. Watson Research Center, P.O. Box 218, Yorktown Heights, New York 10598 (MEGDANI at YKTMV)*. Mr. Megdanis received the B.A. degree in chemistry and physics from SUNY at Potsdam in 1976 and the M.S. degree in physics from Fairleigh Dickinson University in 1980. He has worked at the Liquid Carbonic Corporation and Lederle Laboratory in the field of analytical chemistry. In 1985 he joined the IBM

Thomas J. Watson Research Center, where he worked in the Advanced Silicon Technology Laboratory as a process engineer in the LPCVD, Oxidation and Diffusion Technology group. He currently works in the Digital Communications IC Design group as a circuit designer.

Sudhir Gowda *IBM Research Division, Thomas J. Watson Research Center, P.O. Box 218, Yorktown Heights, New York 10598 (GOWDA at WATSON).* Dr. Gowda received the B.Tech. degree in electrical engineering in 1987 from the Indian Institute of Technology, Madras, India. He received the M.S. and Ph.D. degrees in electrical engineering in 1989 and 1992, respectively, from the University of Southern California at Los Angeles. While at USC, Dr. Gowda was a research assistant at the MOSIS Service of the Information Sciences Institute, Marina del Rey, California, as well as the USC Department of Electrical Engineering. He contributed significantly to the development of the BSIM-plus submicron MOS transistor model for analog and digital VLSI circuits. He also helped develop and teach a senior-level class on VLSI design. In 1992, Dr. Gowda joined the VLSI Design and Communication Technology Department at the IBM Thomas J. Watson Research Center, where he is now a research staff member working on high-speed computing and computer communications. He has published more than twenty technical papers in scientific journals and conferences. Dr. Gowda's research interests include architectural design and detailed circuit implementation for high-speed computers, signal processors, and communication circuits.

Kevin R. Wrenner *IBM Research Division, Thomas J. Watson Research Center, P.O. Box 218, Yorktown Heights, New York 10598 (WREN at YKTVMV).* Mr. Wrenner is an advisory engineer with the Digital Communication IC Design group. He received his B.S. and M. Eng. degrees in electrical engineering from Cornell University in 1986 and 1988, respectively. From 1988 to 1989 he worked for Digital Equipment Corporation, designing digital MOS integrated circuits. In 1989 he joined the IBM Research Division at the Thomas J. Watson Research Center. Since then he has contributed to the design of GaAs and CMOS ASICs for high-speed computer links, and custom CMOS macros for various applications including voltage conversion and signal processing. Mr. Wrenner is a member of Eta Kappa Nu and the Tau Beta Pi Association.